Processing of Text Documents for Subsequent Semantic Analysis

Tina Eliassi-Rad

Center for Applied Scientific Computing

Mike Firpo

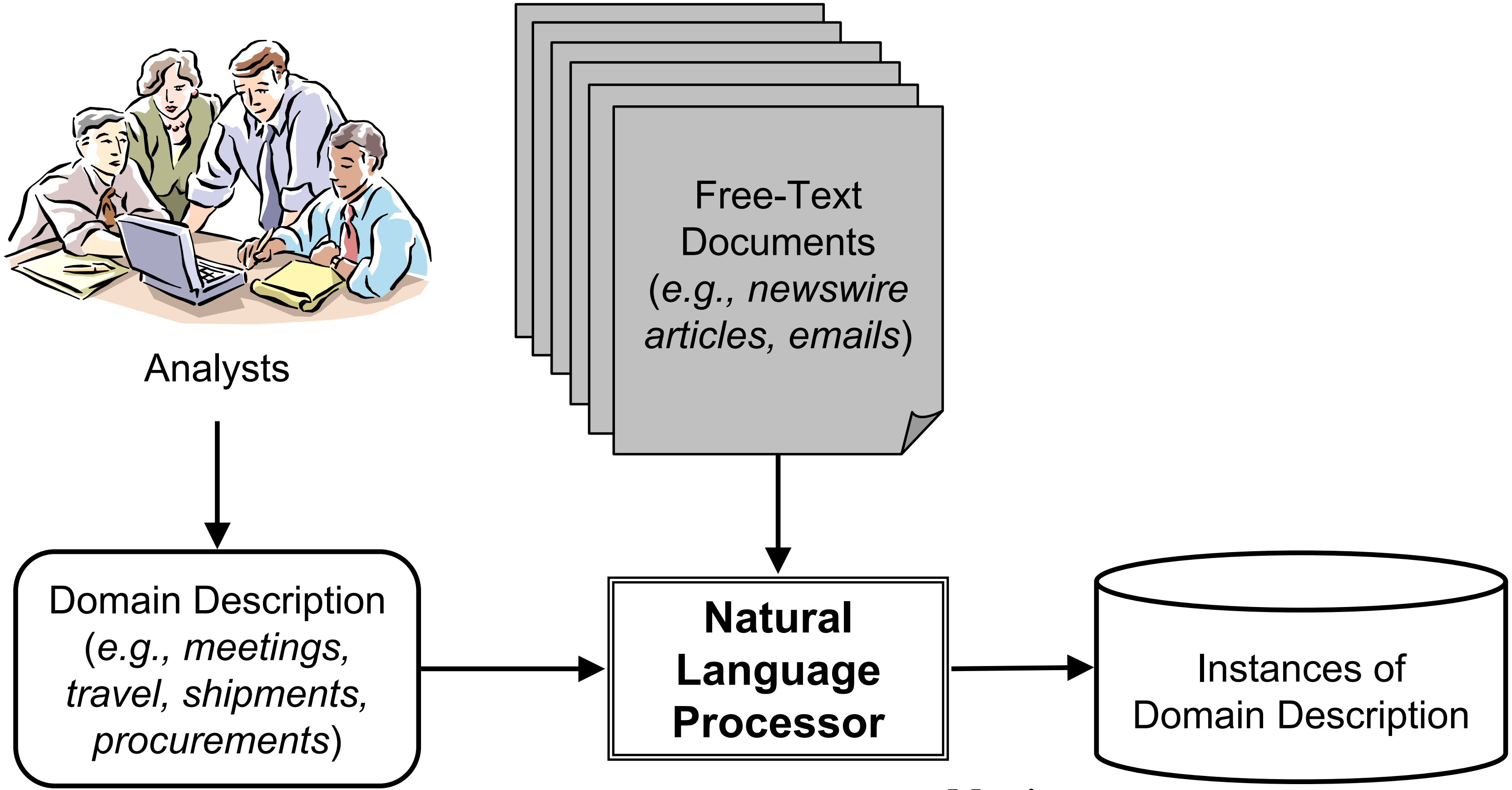
Computing Applications and Research

UCRL-PRES-203099

Presentation for DHS ASC PI Meeting

03/16/2004

Manually reading all available and relevant textual information is daunting!!



Analysts

Free-Text Documents (*e.g., newswire articles, emails*)

Domain Description (*e.g., meetings, travel, shipments, procurements*)

Natural Language Processor

Instances of Domain Description

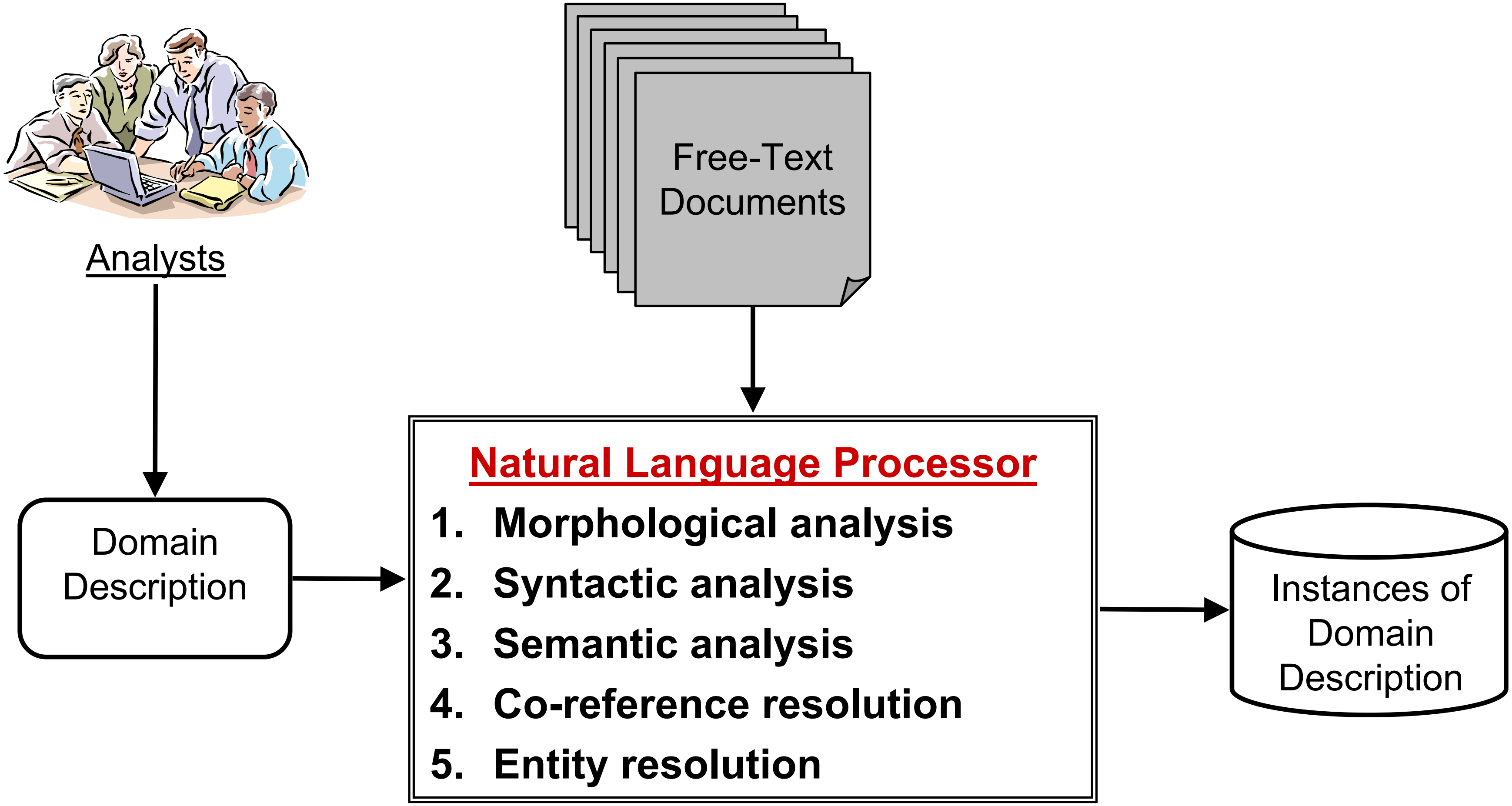
Meeting

***Document ID*:**	**1**
***People*:**	***John Smith, Jane Adams***
***Location*:**	***Livermore, CA***
***Purpose*:**	***DHS***

Relevance to Homeland Security

- Our sponsor is LLNL’s Information Operations Assurance Center (IOAC) .
  - IOAC is part of LLNL’s NAI directorate.
  - Our point of contact is Everett Wheelock.
  - This work helps IOAC in their text analysis tasks by automatically generating semantic data structures from free-text.
- We started this project on February 17, 2004.
  - Mike Firpo: 50%
  - Tina Eliassi-Rad: 5%

Steps in Natural Language Processing for Information Extraction



Analysts

Free-Text Documents

Domain Description

Natural Language Processor

1. **Morphological analysis**
2. **Syntactic analysis**
3. **Semantic analysis**
4. **Co-reference resolution**
5. **Entity resolution**

Instances of Domain Description

Morphological Analysis

- Individual words are analyzed into their components.
  - Example: The word "John's" is pulled apart into the proper noun "John" and the possessive suffix "'s".
- Non-word tokens, such as punctuation, are separated from the words.
  - Example: "John got a 5% raise." becomes "John got a 5 % raise ."

Natural Language Processor

1. Morphological analysis
2. Syntactic analysis
3. Semantic analysis
4. Co-reference resolution
5. Entity resolution

Syntactic Analysis

- **Linear sequences of words are transformed into structures that show how the words relate to each other.**
  - **Example: "I saw Smith." is transformed into (S (NP (PRP I)) (VP (VBD saw) (NP (NNP Smith))) (. .)).**

Natural Language Processor

1. **Morphological analysis**
2. **Syntactic analysis**
3. **Semantic analysis**
4. **Co-reference resolution**
5. **Entity resolution**

Semantic Analysis

- The structures created by the syntactic analyzer are assigned meanings.
  - A mapping is made between the syntactic structures and objects in the task domain.
  - Example: Syntactic analyzer outputs (S (NP (PRP I)) (VP (VBD saw) (NP (NNP Smith))) (. .)). Semantic analyzer transforms this into [S [*AGENT* I] [*MEETING* saw] [*PERSON* Smith] [*PUNCTUATION* .].

Natural Language Processor

1. Morphological analysis
2. Syntactic analysis
3. Semantic analysis
4. Co-reference analysis
5. Entity resolution

Co-reference Resolution

- The meanings of individual sentences that depend on other sentences in the document are resolved.
  - Example: In the text, “I saw Smith. He was with Adams.” the pronoun “He” is resolved to refer to “Smith.”

Natural Language Processor

1. Morphological analysis
2. Syntactic analysis
3. Semantic analysis
4. Co-reference resolution
5. Entity resolution

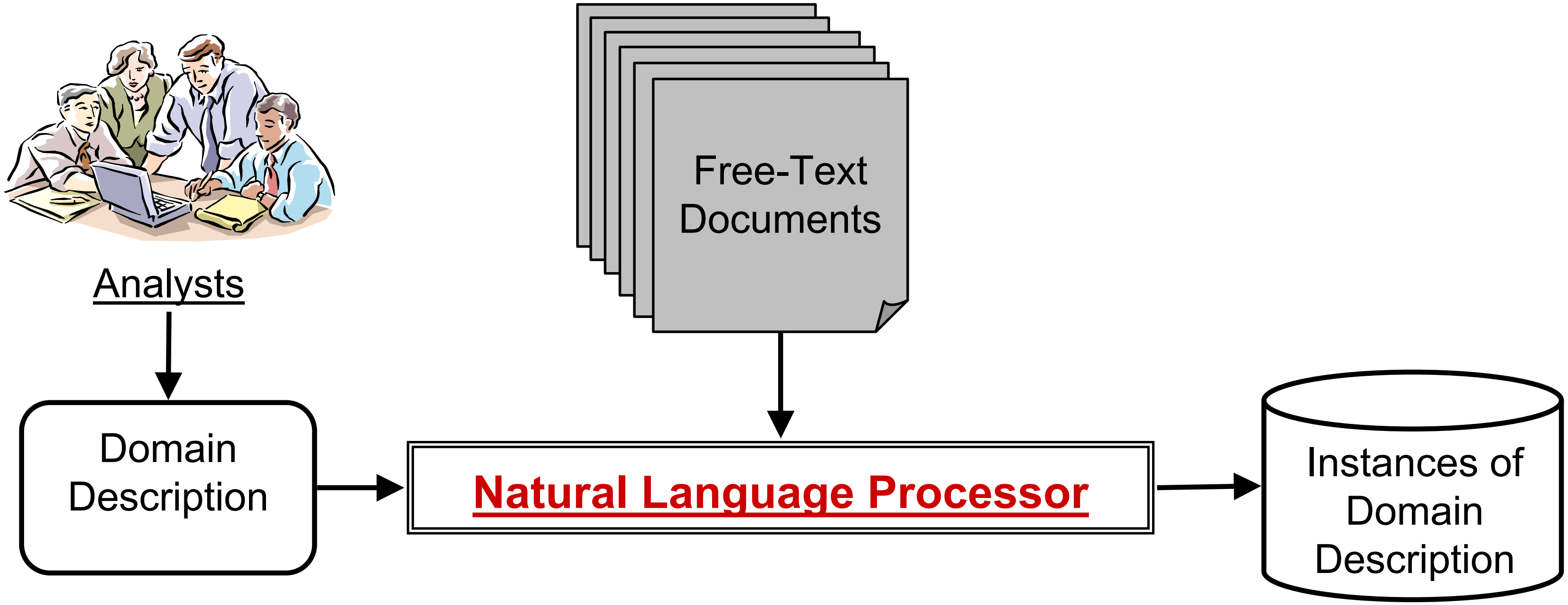
Entity Resolution

- Multiple words/phrases refer to the same entity.
  - Example: In the text, "I saw John Smith. Jane Adams was with John." the name "John Smith" and "Smith" are referring to the same entity.

Natural Language Processor

1. Morphological analysis
2. Syntactic analysis
3. Semantic analysis
4. Co-reference resolution
5. Entity resolution

Evaluation Metrics For Natural Language Processor



Analysts

Free-Text Documents

Domain Description

Natural Language Processor

Instances of Domain Description

- **Accuracy = (TP + TN) / (TP + TN + FP + FN)**
- **Recall = TP / (TP + FN)**
- **Precision = TP / (TP + FP)**
- $\mathbf{F_1 = (2 \times Precision \times Recall) / (Precision + Recall)}$

Accomplishments

- Wrote a pre-processor for converting IOAC’s text documents into meta data (*i.e.*, semi-structured text) and articles (*i.e.*, free-text)
- Examined several existing morphological and syntactic analyzers
  - *Stanford Lexical Parser*
  - *Marmot*
  - *Brill’s Tagger*

Natural Language Processor

1. Morphological analysis
2. Syntactic analysis
3. Semantic analysis
4. Co-reference resolution
5. Entity resolution

Future Plans

- Solve current problems with state-of-the-art syntactic analyzer
  - Example: Handling of long sentences (> 43 words)
- Examination of state-of-the-art semantic analyzers, co-reference solvers, and entity resolution systems
- Solve forthcoming problems with semantic analyzers, co-reference solvers, and entity resolution systems
- Develop a prototype natural language processor for IOAC by October 1, 2004

Natural Language Processor

1. Morphological analysis
2. Syntactic analysis
3. Semantic analysis
4. Co-reference resolution
5. Entity resolution

Conclusion

- **Problem:**
  - Transform free-text documents into a semantic data structures that represents the topics in which analysts are interested
- **Solution:**
  - Implement a natural language processor containing: (1) morphological analyzer, (2) syntactic analyzer, (3) semantic analyzer, (4) co-reference solver, (5) entity resolution system
- **Benefits:**
  - Allows for subsequent discovery of non-trivial, embedded, and novel information in free-text documents

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.1.

eliassi@llnl.gov